

Analyzing Discrete Movement Data as a Correlated Random Walk

BU-950-M*

December 1987
Revised May 1988

C. E. McCulloch and M. L. Cain

Biometrics Unit and Section of Ecology and Systematics
Cornell University, Ithaca, NY 14853

* BU-950-M in the Technical Report Series of the Biometrics Unit, Cornell University, Ithaca, NY 14853.

Analyzing Discrete Movement Data as a Correlated Random Walk

C. E. McCulloch¹ and M. L. Cain²

Biometrics Unit¹ and Section of Ecology and Systematics²

Cornell University, Ithaca, NY 14853

Introduction

To link movement behavior to its many important consequences requires a description and analysis of the process of movement. Common approaches to this problem include statistical description (Angevine and Handel 1986), computer simulation (Jones 1977; Jones *et al.* 1980), and diffusion models (Okubo 1980; Kareiva 1982) of the movement process. Each of these approaches has its strengths and weaknesses, and can be used to advantage in particular situations. In this note we focus on correlated random walk models, a method of analyzing movement data closely related to diffusion models.

Correlated random walk models and diffusion models share the advantage of providing a general framework for making quantitative predictions about an organism's rate of spread. These models also can be used to make succinct comparisons of movement behaviors in different habitats or situations. A primary difference between diffusion and correlated random walk models concerns the time scale at which the movement process is observed. Diffusion models are "continuous time" models: their derivation assumes that there are many moves (steps) within the time period separating sample observations (Levin 1986). In contrast, correlated random walk models are discrete models and are appropriate when the number of steps between observations is small. Much biological movement data, such as the placement of ramets in a plant that spreads vegetatively, or the location of eggs laid by an ovipositing butterfly, are inherently discrete in form.

Kareiva and Shigesada (1983) developed a formula to predict the expected net squared displacement of an organism that followed a correlated random walk. Cain (1988a) extended this approach by developing statistical

tests that allow observed movement data to be compared to random walk predicted displacements. The variances required in these statistical tests were determined by use of bootstrap techniques.

In this paper we present an algorithm that calculates exactly the variance in squared displacement: because our algorithm is more accurate than the bootstrap (see below), its use improves statistical comparisons of observed displacements to random walk predicted displacements. We also provide an approximate formula for the expected net displacement of a correlated random walk, and we use computer simulations to demonstrate the conditions under which our approximation is likely to be accurate. The expected squared displacement, the variance in squared displacement, and the approximation for the net displacement are all calculated from individual movement data that are easy to collect in the field (namely, distributions of move lengths and turning angles). We illustrate the techniques we discuss with examples of insect movement and the vegetative spread of clonal plants.

Calculation of the Variance in Squared Displacement

Kareiva and Shigesada (1983) modeled movement paths as a sequence of straight line moves (Fig. 1), in which move lengths (denoted l_i) and turning angles (denoted θ_i) were chosen from probability distributions $p(l)$ and $g(\theta)$, respectively. Note that turning angles were measured relative to the previous direction of movement (Fig. 1). To derive their formula for the expected squared displacement (R_n^2), Kareiva and Shigesada assumed that there was no correlation within or among turning angles and move lengths, i.e., they assumed that movements could be represented as a series of independent, random draws from the distributions $p(l)$ and $g(\theta)$. Such a

process is termed a correlated random walk, with the distribution $g(\theta)$ determining the degree of correlation that present movements have with the previous direction of movement. In this paper we adopt the model of movement described in Kareiva and Shigesada (1983).

Kareiva and Shesada (1983) present a formula for the squared displacement expected in a correlated random walk; to use the statistical tests in Cain (1988a) we must also calculate the variance of the squared displacement. To calculate $\text{Var}(R_n^2)$ we first calculate $E[R_n^4] = E[(X_n^2 + Y_n^2)^2]$, where X_n and Y_n represent the x- and y-coordinates at step n. These can be found by summing the individual movements:

$$X_n = \sum_{i=1}^n l_i \cos \alpha_i \quad \text{and} \quad Y_n = \sum_{i=1}^n l_i \sin \alpha_i ,$$

where α_i is the angle of the i^{th} movement, measured from the x-axis. Since the turning angles, θ_i , are measured from the previous direction of movement, the angle of the i^{th} movement is the sum of all previous angles:

$$\alpha_i = \alpha + \sum_{j=1}^{i-1} \theta_j \quad (\text{see Figure 1}) .$$

Because it has no effect on calculations concerning R_n , we may assume a random placement of the x-and y-axes without loss of generality. This implies that α , the first angle, has a uniform distribution on $(-\pi, \pi)$ and we will use that fact to simplify calculations. Expanding $E[R_n^4]$ gives the formula

$$\begin{aligned} E[(X_n^2 + Y_n^2)^2] &= E[X_n^4 + 2X_n^2 Y_n^2 + Y_n^4] \\ &= 2E[X_n^4] + 2E[X_n^2 Y_n^2] , \end{aligned}$$

the last equality following from the fact that X_n and Y_n are interchangeable. It remains to calculate $E[X_n^4]$ and $E[X_n^2 Y_n^2]$. Expanding X_n^4 directly and using the independence of the l_i and θ_i , and hence the independence of l_i and α_i , gives

$$E[X_n^4] = \sum_i \sum_j \sum_k \sum_m E[l_i l_j l_k l_m] E[\cos \alpha_i \cos \alpha_j \cos \alpha_k \cos \alpha_m] .$$

Calculation of $E[\lambda_i \lambda_j \lambda_k \lambda_m]$ is straightforward and depends only upon which subscripts are the same. For example, if none of the subscripts are the same then it equals $E[\lambda]^4$, the fourth power of the first moment of the distribution $p(\lambda)$, or if $i=j \neq k=m$ then it equals $E[\lambda^2]^2$. Evaluation of the expected value of the cosine product is more problematic since it depends on whether $i < j$, $i < k$, etc. An outline of the derivation is given in the Appendix.

$E[X_n^2 Y_n^2]$ is evaluated in a similar manner except that the calculation of $E[\cos \alpha_i \cos \alpha_j \sin \alpha_k \sin \alpha_m]$ depends on the ordering of the sines and cosines. Thus, each of the six possible orderings must be handled separately and the appropriate one used when summing.

A program to evaluate $E[R_n^4]$ was written in the matrix language GAUSS (pseudo-code is provided in the Appendix). The program can evaluate $E[R_n^4]$ for $n = 1, 2, \dots, 10$ in approximately 80 minutes, even though no attempt has been made to optimize it for speed. Once $E[R_n^4]$ has been evaluated, the variance of R_n^2 is calculated from the relation $\text{var}(R_n^2) = E[R_n^4] - (E[R_n^2])^2$, where $E[R_n^2]$ is determined from the formula in Kareiva and Shigesada (1983).

We verified our algorithm with extensive computer simulations. For a broad range of gamma-distributed move lengths and circular-normal-distributed turning angles, we simulated movement paths in which the correlated random walk assumptions were satisfied exactly. We then compared the variance of R_n^2 calculated from these simulations to that obtained from our algorithm; in all cases the agreement was close. The algorithm is faster than a simulation since, for some distributions, up to 250,000 movement paths were required for the simulations to be accurate. Gamma distributions with shape parameters less than 2.5 and circular-normal distributions with concentration parameters less than 1.5 required large

numbers of movement paths. For such distributions, our algorithm was roughly 50 times faster than simulations. Of course, another advantage of the algorithm is that one need not be concerned about simulation error and the number of replications required.

An Approximate Formula for the Expected Net Displacement in a Correlated Random Walk

The net squared displacement and the variance in squared displacement can be used to make statistical comparisons between observed displacements and correlated random walk predicted displacements (Cain 1988a). The net squared displacement formula also can be used to estimate the diffusion constant (D), a measure of the long-term rate of spread of populations (Cain 1988b). These and other uses notwithstanding, expected squared displacement is not a biologically intuitive measure: it is, for example, much easier to think in terms of net displacements (e.g., m/day) than net squared displacements (m^2 /day). We have not been able to derive an exact formula for the expected net displacement ($|R_n|$); instead, we present a simple approximation that is accurate in many circumstances.

To derive an approximate formula for $E[|R_n|]$, we regard $|R_n|$ as a function of $S = (X_n^2 + Y_n^2)$ and approximate using a Taylor series expansion of the square root function:

$$\begin{aligned} f(s) = s^{\frac{1}{2}} \approx \mu_S^{\frac{1}{2}} + \left(\frac{\partial}{\partial s} f(s) \Big|_{s=\mu_S} \right) (s - \mu_S) \\ + \frac{1}{2} \left(\frac{\partial^2}{\partial^2 s} f(s) \Big|_{s=\mu_S} \right) (s - \mu_S)^2, \end{aligned} \quad (1)$$

where $\mu_S = E[S]$. Using (1), we approximate the expected value of $|R_n|$ as

$$\begin{aligned}
 E[S^{\frac{1}{2}}] &\approx \mu_S^{\frac{1}{2}} - \frac{1}{8} \mu_S^{\frac{1}{2}} \frac{3}{2} \sigma_S^2 \\
 &= \mu_S^{\frac{1}{2}} \left(1 - \frac{1}{8} \frac{\sigma_S^2}{\mu_S^2} \right) \\
 &= \mu_S^{\frac{1}{2}} \left(1 - \frac{1}{8} CV_S^2 \right), \tag{2}
 \end{aligned}$$

where CV_S = coefficient of variation of S .

Equation (2) shows that the approximate formula gives a first order correction to the naive estimate of $E[S^{\frac{1}{2}}]$, $\mu_S^{\frac{1}{2}}$. To approximate $E[|R_n|]$, we will use $S = (X_n^2 + Y_n^2) = R_n^2$ to get

$$E[|R_n|] \approx E[R_n^2]^{\frac{1}{2}} \left\{ 1 - \frac{1}{8} [\text{Var}(R_n^2)/E[R_n^2]^2] \right\}. \tag{3}$$

We determined the accuracy of this approximation by comparing net displacements calculated from (3) to net displacements obtained by simulating the correlated random walk process (Table 1). In our simulations we used both empirical and theoretical distributions of move lengths and turning angles: the empirical distributions were for *Battus philenor* (pipevine swallowtail, Rausher 1979), *Pieris rapae* (cabbage white butterfly, Root and Kareiva 1984), and *Solidago altissima* (tall goldenrod, Cain 1988b). For the theoretical distributions, move lengths were drawn from gamma distributions and turning angles were drawn from circular normal distributions; the parameter ranges (see Table 1) for these distributions were based on published insect movement data (Kareiva and Shigesada 1983; Cain 1985). As Table 1 indicates, the $|R_n|$ approximation was very accurate for large values of the gamma shape parameter (α). In addition, for fixed values of the shape parameter, the accuracy of the approximation improved considerably as the circular normal concentration parameter (κ) increased

(an increase in κ reduces the spread of the turning angles; see Batschelet 1965). The approximations were accurate for all of the empirical distributions: even for *P. rapae*, the least accurate of the three cases (Table 1), approximate and simulated $|R_n|$ values were in close agreement after the first few moves (Fig. 2).

The accuracy of our approximation is governed by the higher order terms in the Taylor's series, the first of which is

$$\epsilon = E[(S - \mu_S)^3] / 16\mu_S^3 .$$

$E[(S - \mu_S)^3]$ can be expanded to give:

$$E[(S - \mu_S)^3] = E[S^3] - 3\mu_S\sigma_S^2 - \mu_S^3 .$$

For the first of n steps, $E[S^3]$ can be calculated exactly (it equals $E[l^6]$), as can μ_S and σ_S^2 . Hence, for a given distribution of move lengths $p(l)$, we can calculate ϵ and use the result as a measure of the likely accuracy of the $|R_n|$ approximation: large values of ϵ suggest that the approximation will not be accurate. For all 15 distributions tested, this method provided a good indication of the accuracy of the $|R_n|$ approximation. For example, whenever ϵ was less than one, the maximum percent deviation of the approximate $|R_n|$ values from the expected values was 13.2%; more commonly deviations were near or below 1.0% (compare Tables 1 and 2). For the first step, the accuracy of the approximation also can be evaluated by calculating $|R_n|$ exactly and comparing it to the value determined by (3).

Discussion

In this note we describe an algorithm that uses individual movement data to calculate the variance in squared displacement for a correlated random walk process. This variance then can be used to compare observed and correlated random walk predicted displacements (see Cain 1988a for a

description of statistical tests). The variance in squared displacement also can be determined by using computer simulations: our algorithm, however, is faster and more accurate.

When statistical analyses support the correlated random walk model, succinct comparisons of an organism's movement behavior in different habitats or situations can be made using predicted net squared displacements and/or long-term diffusion rates (Kareiva and Shigesada 1983; Cain 1988b). However, these quantities are not biologically intuitive measures of displacement: it is, for example, considerably easier to attach biological meaning to net displacements than to net squared displacements. To address this difficulty, we developed a simple approximation to the net displacement expected in a correlated random walk. Our approximation is accurate in many circumstances, but should not be used when the first step error term (described above) is large. In such cases, predicted net displacements can be obtained by computer simulation.

Acknowledgments

We thank Mark Andersen, Simon Levin, and an anonymous reviewer for helpful comments on an early version of this paper. This work was supported in part by McIntire-Stennis Grant NYC-183568 to Simon Levin.

APPENDIX

To simplify the calculations of $E[X_n^4]$, let $\alpha_{[i]}$ denote the values of α_i ordered by their *subscripts*, i.e., $\alpha_{[1]}$ represents the α_i with the smallest subscript. Furthermore, let

$$\eta = \alpha_{[1]}$$

$$\Delta_1 = \alpha_{[2]} - \alpha_{[1]}$$

$$\Delta_2 = \alpha_{[3]} - \alpha_{[2]}$$

$$\Delta_3 = \alpha_{[4]} - \alpha_{[3]} .$$

Note that η has a uniform distribution on $(-\pi, \pi)$, since α does, and that η and the Δ_i are independent since they are functions of different θ_i . Some or all of the Δ_i may be zero. As noted in the text, the main complication is the calculation of the cosine product which is equal to

$$E[\cos(\eta)\cos(\eta+\Delta_1)\cos(\eta+\Delta_1+\Delta_2)\cos(\eta+\Delta_1+\Delta_2+\Delta_3)] .$$

Using standard trigonometric identities, this can be rewritten as

$$\begin{aligned} & \frac{1}{4}E[\cos(\Delta_1)\cos(\Delta_2) + \cos(\Delta_1)\cos(2\eta+2\Delta_1+2\Delta_2+\Delta_3) \\ & + \cos(\Delta_3)\cos(2\eta+\Delta_1) + \frac{1}{2}\cos(\Delta_1+2\Delta_2+\Delta_3) \\ & + \frac{1}{2}\cos(4\eta+3\Delta_1+2\Delta_2+\Delta_3)] . \end{aligned}$$

This simplifies considerably due to the uniform distribution of η . For example, if we use U to denote a random variable with a uniform distribution on $(-\pi, \pi)$,

$$\begin{aligned} E[\cos(\Delta_3)\cos(2\eta+\Delta_1)] &= E[\cos(\Delta_3)]E[\cos(2\eta+\Delta_1)] \\ &= E[\cos(\Delta_3)]E[\cos(U)] \\ &= E[\cos(\Delta_3)] \cdot 0 \\ &= 0 . \end{aligned}$$

Using similar calculations, we can simplify $E[X_n^4]$ to $\frac{1}{8} \left(E[2\cos(\Delta_1)\cos(\Delta_3)] + E[\cos(\Delta_1+2\Delta_2+\Delta_3)] \right)$. Using cosine addition formulas, this can be expanded to equal

$$\begin{aligned} \frac{1}{8} \left(2E[\cos\Delta_1]E[\cos\Delta_3] + E[\cos\Delta_1]E[\cos2\Delta_2]E[\cos\Delta_3] \right. \\ \left. - E[(\sin\Delta_1)E[\sin2\Delta_2]E[\cos\Delta_3] - E[\cos\Delta_1]E[\sin2\Delta_2]E[\sin\Delta_3]] \right. \\ \left. - E[\sin\Delta_1]E[\cos2\Delta_2]E[\sin\Delta_3] \right) . \end{aligned}$$

To finalize the calculation of $E[X_n^4]$, it remains to evaluate the terms like $E[\sin\Delta_1] = E[\sin(\alpha_{[2]} - \alpha_{[1]})] = E \left[\sin \left(\sum_{j=\ell}^m \theta_j \right) \right]$. Now, denoting $\sqrt{-1}$ by i ,

$$\begin{aligned} E[\sin\Delta_1] &= E \left[\frac{1}{2i} \left\{ \exp \left(i \sum_{j=\ell}^m \theta_j \right) - \exp \left(-i \sum_{j=\ell}^m \theta_j \right) \right\} \right] \\ &= \frac{1}{2i} \left(\prod_{j=\ell}^m E \left[e^{i\theta_j} \right] - \prod_{j=\ell}^m E \left[e^{-i\theta_j} \right] \right) \\ &= \frac{1}{2i} \left((c+is)^k - (c-is)^k \right) , \end{aligned}$$

where

$$c = E[\cos\theta_j], \quad s = E[\sin\theta_j] \quad \text{and} \quad k = m-\ell+1 .$$

Note that the result only depends on the difference in the indices m and ℓ , not on their actual value. Expanding using the binomial formula we can finally write, for c and s nonzero,

$$E[\sin\Delta_1] = \frac{-c^{k+1}k!}{s} \sum_{j=1}^{\left[\frac{k+1}{2} \right]} \left(\frac{-s^2}{c^2} \right)^j / (k-2j+1)!(2j-1)! ,$$

where $\lceil \cdot \rceil$ is the greatest integer function. $E[\cos\Delta]$, $E[\cos2\Delta]$ and $E[\sin2\Delta]$ can be evaluated similarly. While not pretty, $E[X_n^4]$ can be straightforwardly evaluated on a computer given values of $E[\ell]$, $E[\ell^2]$, $E[\ell^3]$, $E[\ell^4]$, $E[\sin\theta]$, $E[\cos\theta]$, $E[\sin2\theta]$ and $E[\cos2\theta]$ and using the above calculations.

Pseudo-code for the program is given below. The variables and notation are the same as in the text.

```

MAIN PROGRAM    {computes  $E[R_n^k]$ }

SUBPROGRAMS    C(k), S(k), C2(k), S2(k), FOURL(i, j, k, m), FOURCOS(i, j, k, m),
                SINCOS(i, j, k, m)

INPUT          n,  $E[\lambda]$ ,  $E[\lambda^2]$ ,  $E[\lambda^3]$ ,  $E[\lambda^4]$ ,  $E[\cos\theta]$ ,  $E[\sin\theta]$ ,
                 $E[\cos 2\theta]$  and  $E[\sin 2\theta]$  .

COMPUTE        C(k) =  $E\left[\cos\left(\sum_{i=m}^{m+k} \theta_i\right)\right]$ 

                S(k) =  $E\left[\sin\left(\sum_{i=m}^{m+k} \theta_i\right)\right]$ 

                C2(k) =  $E\left[\cos\left(2 \sum_{i=m}^{m+k} \theta_i\right)\right]$ 

                S2(k) =  $E\left[\sin\left(2 \sum_{i=m}^{m+k} \theta_i\right)\right]$ 

                for k = 0, 1, ..., n-1 .

COMPUTE         $\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \sum_{m=1}^n \text{FOURL}(i, j, k, m) * 2 * [\text{FOURCOS}(i, j, k, m)$ 
                 $+ \text{SINCOS}(i, j, k, m)]$  .

END MAIN

SUBPROGRAM      C(k)    {computes  $E\left[\cos\left(\sum_{i=m}^{m+k} \theta_i\right)\right]$ }

LET            c =  $E[\cos \theta]$ 
                s =  $E[\sin \theta]$ 

RETURN         $\sum_{m=0}^{\left[\frac{k}{2}\right]} \binom{k}{2m} c^{k-2m} s^{2m} (-1)^m$ 

END            C

```

```

SUBPROGRAM   C(k)   {computes  $E\left[\sin\left(\sum_{i=m}^{m+k} \theta_i\right)\right]}$ 

      LET      c = E[cos $\theta$ ]
              s = E[sin $\theta$ ]

      RETURN    $\sum_{m=1}^{\left[\frac{k+1}{2}\right]} \binom{k}{2m-1} c^{k-2m+1} s^{2m-1} (-1)^{m-1}$ 

END          S

```

Subprograms C2 and S2 are identical to C and S, except using $c = E[\cos 2\theta]$ and $s = E[\sin 2\theta]$.

```

SUBPROGRAM   FOURL(i,j,k,m)   {computes  $E[\rho_i \rho_j \rho_k \rho_n]}$ 
ORDER        i,j,k,m   so that  $i \leq j \leq k \leq m$ 

      IF      i < j < k < m           RETURN   E[ $\rho$ ]4
      ELSE IF
            i < j < k = m   OR
            i < j = k < m   OR
            i = j < k < m           RETURN   E[ $\rho^2$ ]E[ $\rho$ ]2
      ELSE IF
            i = j < k = m           RETURN   E[ $\rho^2$ ]2
      ELSE IF
            i < j = k = m   OR
            i = j = k < m           RETURN   E[ $\rho$ ]E[ $\rho^3$ ]
      ELSE
            RETURN   E[ $\rho^4$ ]

END          FOURL

```

```

SUBPROGRAM   FOURCOS(i,j,k,m)   {computes  $E[\cos \alpha_i \cos \alpha_j \cos \alpha_k \cos \alpha_m]}$ 
ORDER        i,j,k,m   so that  $i \leq j \leq k \leq m$ 
COMPUTE      D1 = j-i
              D2 = k-j
              D3 = m-k

      RETURN   [ 2*C(D1)*C(D3) - C(D1)*C2(D2)*C(D3)
                + S(D1)*S2(D2)*C(D3) + S(D1)*C2(D2)*S(D3)
                + C(D1)*S2(D2)*S(D3) ]/8

END          FOURCOS

```

```

SUBPROGRAM   SINCOS(i,j,k,m)    {computes  $E[\cos\alpha_i \cos\alpha_j \sin\alpha_k \sin\alpha_m]$ }

ORDER       i,j,k,m    so that    $i \leq j$     and    $k \leq m$ 

LET          $i_1, i_2, i_3, i_4$  be the ordered values of  $i, j, k, m (i_1 \leq i_2 \leq i_3 \leq i_4)$ 

COMPUTE     D1 =  $i_2 - i_1$ 
            D2 =  $i_3 - i_2$ 
            D3 =  $i_4 - i_3$ 

IF           $i \leq k$     AND     $j \leq k$ 

RETURN      [  $2 * C(D1) * C(D3) - C(D3) * C2(D2) * C(D3)$ 
              +  $S(D1) * S2(D2) * C(D3) + S(D1) * C2(D2) * S(D3)$ 
              +  $C(D1) * S2(D2) * S(D3) ] / 8$  .

ELSE IF      $i \leq k$     AND     $j > k$     AND     $j \leq l$ 

RETURN      [  $2 * S(D1) * S(D3) + C(D1) * C2(D2) * C(D3)$ 
              -  $S(D1) * S2(D2) * C(D3) - S(D1) * C2(D2) * S(D3)$ 
              -  $C(D1) * S2(D2) * S(D3) ] / 8$  .

ELSE IF      $i \leq k$     AND     $j > k$     AND     $j > l$ 

RETURN      [  $-2 * S(D1) * S(D3) + C(D1) * C2(D2) * C(D3)$ 
              -  $S(D1) * S2(D2) * C(D3) - S(D1) * C2(D2) * S(D3)$ 
              -  $C(D1) * S2(D2) * S(D3) ] / 8$  .

ELSE IF      $i > k$     AND     $i \geq l$ 

RETURN      [  $2 * C(D1) * C(D3) - C(D1) * C2(D2) * C(D3)$ 
              +  $S(D1) * S2(D2) * C(D3) + S(D1) * C2(D2) * S(D2)$ 
              +  $C(D1) * S2(D2) * S(D3) ] / 8$  .

ELSE IF      $i > k$     AND     $i < l$     AND     $j \leq l$ 

RETURN      [  $-2 * S(D1) * S(D2) + C(D1) * C2(D2) * C(D3)$ 
              -  $S(D1) * S2(D2) * C(D3) - S(D1) * C2(D2) * S(D3)$ 
              -  $C(D1) * S2(D2) * S(D3) ] / 8$  .

ELSE

RETURN      [  $2 * S(D1) * S(D3) + C(D1) * C2(D2) * C(D3)$ 
              -  $S(D1) * S2(D2) * C(D3) - S(D1) * C2(D2) * S(D3)$ 
              -  $C(D1) * S2(D2) * S(D3) ] / 8$  .

END         SINCOS

```

LITERATURE CITED

- Angevine, M. W. and S. N. Handel (1986). Invasion of forest floor space, clonal architecture, and population growth in the perennial herb *Clintonia borealis*. *Journal of Ecology* **74**: 547-560.
- Batschelet, E. (1965). Statistical methods for the analysis of problems in animal orientation and certain biological rhythms. Monograph, American Institute of Biological Sciences, Washington, D.C.
- Cain, M. L. (1988a). Movement pattern analysis: statistical evaluation of correlated random walk models. *Ecology*, in review.
- Cain, M. L. (1988b). The process of clonal growth in *Solidago altissima*. *Journal of Ecology*, in review.
- Cain, M. L. (1985). Random search by herbivorous insects: a simulation model. *Ecology* **66**: 876-888.
- Jones, R. E. (1977). Movement patterns and egg distributions in cabbage butterflies. *Journal of Animal Ecology* **46**: 195-212.
- Jones, R. E., N. Gilbert, M. Guppy, and V. Nealis (1980). Long-distance movement of *Pieris rapae*. *Journal of Animal Ecology* **49**: 629-642.
- Kareiva, P. M. (1982). Experimental and mathematical analyses of herbivore movements: quantifying the influence of plant spacing and quality on foraging discrimination. *Ecological Monographs* **52**: 261-282.
- Kareiva, P. M. and N. Shigesada (1983). Analyzing insect movement as a correlated random walk. *Oecologia* **56**: 234-238.

- Levin, S. A. (1986). Random walk models of movement and their implications.
In *Mathematical Ecology*, T. G. Hallam and S. A. Levin, eds.
Springer-Verlag, Berlin, pp. 149-154.
- Okubo, A. (1980). *Diffusion and Ecological Problems: Mathematical Models*.
Springer-Verlag, New York.
- Rausher, M. (1979). Co-evolution in a simple plant-herbivore system.
Dissertation. Cornell University, Ithaca, New York.
- Root, R. B. and P. M. Kareiva (1984). The search for resources by cabbage
butterflies (*Pieris rapae*): ecological consequences and adaptive signi-
ficance of Markovian movements in a patchy environment. *Ecology* 65:
147-165.

Table 1: Percent deviation of approximate $|R_n|$ values from expected $|R_n|$ values. Approximate $|R_n|$ values were determined from Eqn. (3). For the theoretical distributions, expected $|R_n|$ values were determined by stochastic simulations in which move lengths were gamma distributed (α as below, $\beta=1.0$), and turning angles were circular normal distributed ($\kappa=0.5$ for this table; $\kappa=2.0$ and $\kappa=5.0$ also were tested). For the empirical distributions, expected values were determined by stochastic simulations in which move length and turning angle distributions were based on observed data.

<u>move</u>	<u>Theoretical Distributions</u>				<u>Empirical Distributions</u>		
	$\alpha=.5$	$\alpha=2.0$	$\alpha=3.5$	$\alpha=5.0$	<i>P. rapae</i>	<i>B. philenor</i>	<i>S. altissima</i>
1	157.6	13.2	4.6	2.1	38.3	3.6	0.2
2	56.1	5.6	1.7	0.3	12.2	1.3	0.4
3	32.3	3.4	1.0	0.0	6.4	0.4	0.3
4	22.6	2.6	0.7	0.1	3.9	0.3	0.1
5	17.4	2.0	0.6	0.1	2.8	0.6	0.1
6	14.3	1.8	0.7	0.0	1.9	0.5	0.3
7	12.1	1.7	0.7	0.1	1.5	0.4	0.5
8	10.6	1.5	0.7	0.1	1.1	0.5	0.7
9	9.5	1.5	0.7	0.4	0.9	0.5	0.9
10	8.6	1.4	0.8	0.5	0.6	0.5	1.0

Table 2: Magnitude of the error term ϵ for the theoretical and empirical distributions listed in Table 1.

	<u>Distribution</u>	<u>Error Term (ϵ)</u>
Theoretical:		
	$\gamma(0.5,1)$	22.00
	$\gamma(2.0,1)$	0.96
	$\gamma(3.5,1)$	0.27
	$\gamma(5.0,1)$	0.13
Empirical:		
	<i>P. rapae</i>	2.46
	<i>B. philenor</i>	0.22
	<i>S. altissima</i>	0.07

FIGURE LEGENDS

Fig. 1. The method of recording movement parameters for straight line approximations to actual movement paths. Move lengths (l_1) are recorded as net displacements between an organism's successive locations. Turning angles (θ_1) are recorded from $-\pi$ to π , with right turns being negative (θ_2 is thus a negative angle). See the text for a description of α and α_1 .

Fig. 2. Expected (■—■) and approximate (◆ ◆ ◆) net displacements for *Solidago altissima* move length and turning angle frequency distributions. Expected displacements were determined by stochastic simulation; approximate displacements were calculated from Eqn. 3. This graph represents the least accurate approximation of the three data sets examined.

Fig. 1

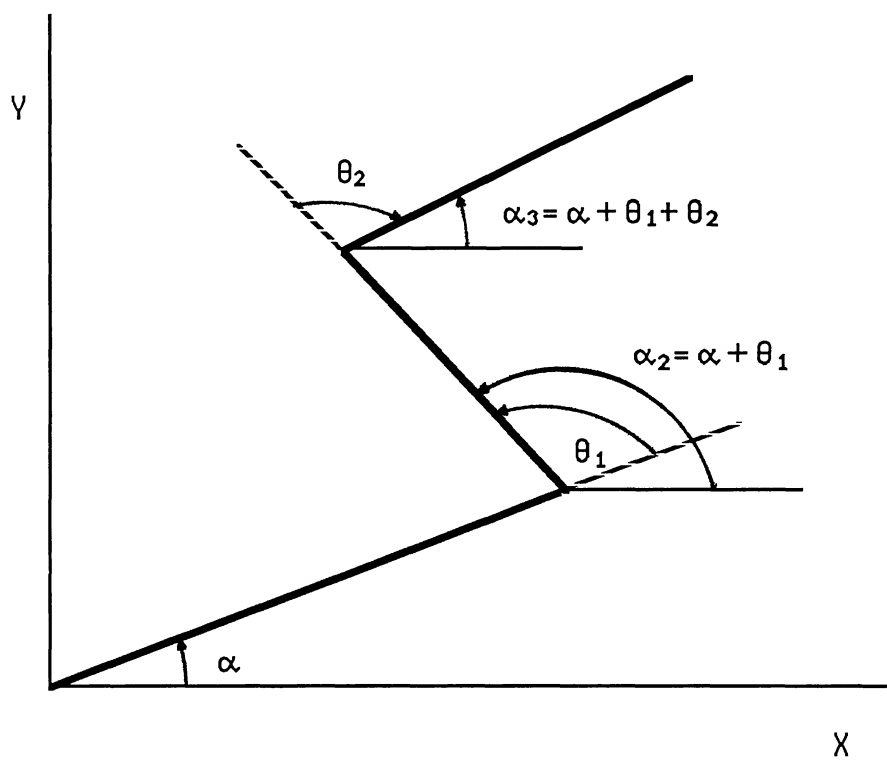


Fig. 2

